Claims

[c1] YOR920040078US121

1. A system for memory management, the system comprising a tag cache in communication with one or more cache devices in a storage hierarchy, wherein:
the tag cache includes tags of recently accessed memory blocks, each tag corresponding to one of the memory blocks and each tag including tag contents;
the tag contents control which memory lines of the corresponding memory block are prefetched into at least one of the cache devices;
the tag contents are updated using a selected subset of processor references, said subset referred to as filtered references; and
the tag contents are modified probabilistically at selected times or events.

[c2] 2.The system of claim 1 wherein the tag contents further control which next virtual memory block is prefetched into at least one of the cache devices.

[c3] 3.The system of claim 1 wherein the tag contents include a memory block real address and one bit for every memory line in the memory block, said bits referred to as

prefetch bits.

[c4] 4.The system of claim 3 wherein the tag contents further include a bit to control prefetching of memory lines from a next virtual memory block, said bit referred to as a next virtual memory block bit.

[c5] 5.The system of claim 4 wherein the next virtual memory block bit in a preceding memory block in a virtual address space is set to a prefetch status when the preceding memory block tag is in the tag cache.

[c6] 6.The system of claim 4 wherein the next virtual memory block bit is turned to a nonprefetch status with a specified probability on certain events.

[c7] 7.The system of claim 6 wherein the certain events include eviction of the tag from the tag cache.

[c8] 8.The system of claim 3 wherein:
one of the cache devices is a level two cache device and another is a level one cache device;
the prefetch bits are set by a selected subset of misses from the level one cache device; and
the misses include at least one miss from the level two cache device.

[c9] 9.The method of claim 3 wherein the tag contents in-

clude 32 prefetch bits.

[c10] 10.The system of claim 3 wherein the value of each prefetch bit determines whether the corresponding memory line should be prefetched.

[c11] 11.The system of claim 3 wherein each of the prefetch bits is reset to a non-prefetch status with a selected probability of between zero and one.

[c12] 12.The system of claim 11 wherein the selected probability is one eighth.

[c13] 13.The system of claim 1 wherein an access by one of the cache devices to a memory line corresponding to one of the prefetch bits results in the prefetch bit not being reset to a nonprefetch status before the tag is stored back to the memory device.14.The system of claim 1 wherein the tag contents further control the sending of the entries from the one or more cache devices to storage hierarchies of other processors.

[c14] 15.The system of claim 1 wherein each of the memory blocks is four thousand and ninety-six (4K) bytes, each of the memory blocks includes 32 memory lines, and each of the memory lines is 128 bytes.

[c15] 16.The system of claim 1 wherein the prefetch occurs

when a filtered reference event occurs and the tag cache does not contain an entry for the corresponding memory block.

[c16] 17.The system of claim 1 wherein an access by one of the cache devices to one of the memory lines results in the prefetch bit corresponding to the memory line not being reset to a nonprefetch status before the tag is stored back to the memory device if the access passes filtering criteria.

[c17] 18.The system of claim 1 wherein the tags are returned to the memory device when the tag is deleted from the tag cache.

[c18] 19.The system of claim 1 wherein one of the cache devices is a level one cache device.

[c19] 20.The system of claim 1 wherein one of the cache devices is a level two cache device.

[c20] 21.The system of claim 1 wherein the memory block is a system page.

[c21] 22.A method for memory management, the method comprising:
receiving a notification of a cache fault from a cache device, the notification including a fault memory block and

a fault memory line;
determining if a tag corresponding to the fault memory block is present in a tag cache, wherein the tag includes prefetch bits corresponding to memory lines contained in a memory block specified by the tag;
in response to not locating the tag corresponding to the fault memory block in the tag cache:
fetching the tag corresponding to the fault memory block into the tag cache;
prefetching the memory lines corresponding to the prefetch bits in the tag that are set to a prefetch status, said prefetching into the cache device; and
resetting each of the prefetch bits which were set to a prefetch status to a nonprefetch status with a selected probability; and
setting the prefetch bit corresponding to the fault memory line in the tag to the prefetch status.

[c22] 23.The method of claim 22 wherein the tag further includes a next virtual memory block bit and the method further comprises prefetching selected lines of the next virtual memory block into the cache device in response to:
not locating the tag corresponding to the fault memory block and the next virtual memory block in the tag cache; and

the next virtual memory block bit being set to a prefetch status.

[c23] 24.The method of claim 23 wherein the prefetching selected lines of the next virtual memory block includes prefetching the memory lines corresponding to the prefetch bits set to a prefetch status in the tag corresponding to the next virtual memory block.

[c24] 25.The method of claim 23 wherein the next vitural memory block bit in a preceeding memory block in a virtual address space is set to a prefetch status when the preceeding memory block tag is in the tag cache.

[c25] 26.The method of claim 23 wherein the next virtual memory block bit is turned to a nonprefetch status with a specified probability on certain events.

[c26] 27.The method of claim 22 wherein the selected probability is one eighth.

[c27] 28.The method of claim 22 wherein an access by the cache device to one of the memory lines results in the prefetch bit corresponding to the memory line not being reset to a nonprefetch status before the tag is stored back to the memory device.

[c28] 29.The method of claim 22 wherein an access by the

cache device to one of the memory lines results in the prefetch bit corresponding to the memory line not being reset to a nonprefetch status before the tag is stored back to the memory device if the access passes filtering criteria.

[c29] 30.The method of claim 22 further comprising returning one of the tags to the memory device when the tag is deleted from the tag cache.

[c30] 31.The method of claim 22 wherein the memory block is a system page.

[c31] 32.A computer program product for cache memory management, the computer program product comprising:
a storage medium readable by a processing circuit and storing instructions for execution by the processing circuit for performing a method comprising:
receiving a notification of a cache fault from a cache device, the notification including a fault memory memory block and a fault memory line;
determining if a tag corresponding to the fault memory block is present in a tag cache, wherein the tag includes a prefetch bit corresponding to memory lines contained in a memory block specified by the tag;
in response to not locating the tag corresponding to the fault memory memory block in the tag cache:

fetching the tag corresponding to the fault memory block into the tag cache;

prefetching the memory lines corresponding to the prefetch bits in the tag that are set to a prefetch status, said prefetching into the cache device; and

resetting each of the prefetch bits which were set to a prefetch status to a nonprefetch status with a selected probability; and

setting the prefetch bit corresponding to the fault memory line in the tag to a prefetch status.